REMARKS

Reconsideration of the above-identified patent application in view of the amendments above and the remarks following is respectfully requested.

Claims 1-49 are in this case. Claims 38-49 were withdrawn by the Examiner from consideration as drawn to a non-elected invention. Claims 1, 17, 33 and 37 have been rejected under § 102(e). Claims 2-16, 18-32 and 34-36 have been rejected under § 103(a).

The claims before the Examiner are directed toward adapters for managing transport service instances, such as queue pairs, that send data packets from a host to a network and receive data packets for the host from the network, and methods of their use. A host interface gets context information about the service instances from a system memory of a host processor. Packet processing circuitry processes data packets that are sent to the network and that are received from the network via a network/fabric interface, according to the context information. Context information that is used frequently by the processing circuitry is kept in a cache.

§ 102(e) Rejections – Starr et al. '581

The Examiner has rejected claims 1, 17, 33 and 37 under § 102(e) as being anticipated by Starr et al., US Patent No. 6,807,581 (henceforth, "Starr et al. '581"). The Examiner's rejection is respectfully traversed.

Starr et al. '581 teach an intelligent network interface card (INIC) (e.g., INIC **22** of Figure 1) that relieves the CPU of a host (e.g. CPU **30** of host **20** of Figure 1) of the burden of network communication protocol processing. For each message that is to be exchanged between the host and a remote host, a communication control block

(CCB) is created. The CCBs are stored in a CCB cache in the INIC (e.g. in CCB cache **74** of INIC **22** of Figure 1).

Comparing the teachings of Starr et al. '581, as illustrated in Figure 1 of that patent, reveals the following correspondences to the elements of the present invention:

PHY **58** and MAC **60** correspond to the network interface of the present invention.

I/O bridge **50** corresponds to the host interface of the present invention.

CPU **30** corresponds to the host processor of the present invention.

Host storage unit **66** corresponds to the system memory of the present invention.

Processor **44** corresponds to the packet processing circuitry of the present invention.

CCB cache **74** corresponds to the cache memory of the present invention, because the CCBs correspond to the context information of the present invention. That the CCBs of Starr et al. '581 correspond to the context information of the present invention is abundantly clear from a comparison of the examples of context information given in the specification (page 3 lines 17-20: destination address, negotiated operating limits, service level, keys for access control; page 3 line 21: current packet sequence number) to the examples of the contents of a CCB that are given by Starr et al. '581 (column 7 lines 30-31: source and destination addresses and ports; column 7 lines 32-36: source and destination MAC addresses, source and destination IP addresses, source and destination TCP ports, TCP variables such as timers and receive and transmit windows for sliding window protocols; column 7 lines 60-63: state information regarding the message, such as the length of the message and the number and order of packets that have been processed).

One crucial difference between the teachings of Starr et al. '581 and the present invention is that the present invention, as recited in claims 1, 17, 33 and 37, stores the context information *both* in the system memory *and* in the cache memory. Indeed, the problem solved by the present invention is that, with "16 million QPs allowed by the IB specification" (page 4 lines 15-16), most of the context information of the present invention *must* be stored in the system memory. By contrast, Starr et al. '581 store the CCBs *only* in the CCB cache and *not* in the host storage unit.

Thus, the present invention, as recited in claims 1, 17, 33 and 37, is not anticipated by Starr et al. '581. Furthermore, the present invention, as recited in claims 1, 17, 33 and 37, is not even obvious from Starr et al. '581. There is neither a hint nor a suggestion in Starr et al. '581 of any need to store CCBs in the host storage unit. Hence, claims 1, 17, 33 and 37 are allowable in their present form over the prior art cited by the Examiner.

§ 103(a) Rejections – Starr et al. '581 in view of Dobbins et al. '123

The Examiner has rejected claims 2-16, 18-32 and 34-36 under § 103(a) as being unpatentable over Starr et al. '581 in view of Dobbins et al., US Patent No. 5,509,123 (henceforth, "Dobbins et al. '123"). The Examiner's rejection is respectfully traversed.

It is demonstrated above that independent claims 1, 17 and 33 are allowable in their present form. It follows that claims 2-16, 18-31 and 34-36, that depend therefrom, also are allowable.

Although claims 2-16, 18-32 and 34-36 are allowable merely by virtue of depending from claims 1, 17 and 33, there are additional reasons why some of these claims are allowable over the prior art cited by the Examiner. Specifically, the Examiner has argued that the features of the present invention that are recited in claims 2, 10 and 18 are taught by Dobbins et al. '123. The following rebuttal of the Examiner's argument was presented in the response to the Office Action mailed August 26, 2004. It is repeated here for the Examiner's convenience.

Dobbins et al. '123 teach an object-oriented architecture for a network router. According to the OSI model, a network router works in layer 3, the "network layer". Architecture **200** of routers **105**, **106**, **107** and **108** includes a forwarding engine **203** that in turn includes forwarding engine objects such as host forwarding engine object **232** and protocol forwarding engine object **234**. Because deciding how to forward a packet along the network is computationally intensive, forwarding engine **203** uses a cache to save forwarding strategies for subsequent packets that share the same source and destination addresses as the packets for which the forwarding strategies were computed. The forwarding strategies are indexed according to destination addresses (column 7 line 62).

How Dobbins et al. '123 index their cache is described in column 7 line 62. The index of the information in the cache for a data packet is the destination address of the data packet. By contrast, the indices of the present invention, as recited in claims 2 and 18, are portions of the service instance numbers.

Turning now to the direction of data flow with respect to the cache, in Dobbins et al. '123, this data flow is only into the cache. The Examiner cited Dobbins et al. '123 column 7 line 67 as anticipating claim 10, which recites fields in the cache memory that are '"updated by the packet processing circuitry" and then "copied back to the context information in the system memory", *i.e.*, data flow from the cache back to the host. The full citation fro Dobbins et al. '123 is column 7 lines 64-67:

> ...if the destination network address is not located in cache memory, accessing a forward look-up table **20** for the best route to the

> destination network address, and then updating its cache. (emphasis added)

The cache in question clearly is the cache of the forwarding engine, so that the data flow is from the host to the cache.

Amendments to the Specification

An inadvertent grammatical error on page 19 line 22 has been corrected.

No new matter has been added.

In view of the above amendments and remarks it is respectfully submitted that independent claims 1, 17, 33 and 37, and hence dependent claims 2-16, 18-32 and 34-36 are in condition for allowance. Prompt notice of allowance is respectfully and earnestly solicited.

Respectfully submitted,

Mark M. Friedman
Attorney for Applicant
Registration No. 33,883

Date: September 5, 2005